

Vrije Universiteit Amsterdam



Universiteit van Amsterdam



Master Thesis

---

# Deep Learning-Enhanced Segmentation and Tracking of E. coli in Time-Lapse Microscopy

---

**Author:** Lucas Reed (2796368)

*Supervisor:* Johan Van Heerden  
*Daily supervisor:* Philipp Savakis  
*Examiner:* Douwe Molenaar

*A thesis submitted in fulfillment of the requirements for  
the joint VU-UvA Master of Bioinformatics and Systems Biology*

August 7, 2025

---

## DECLARATION OF AI USAGE

I hereby acknowledge that the research reported in this thesis has been carried out with the responsible and ethical utilization of artificial intelligence (AI) technologies. AI tools such as ChatGPT- 4 and Claude were utilized to assist in specific tasks, including:

- **Language and Grammar Assistance:** AI-based writing assistants were used to enhance the clarity and coherence of the text, ensuring proper grammar and style.
- **Learning:** AI-powered resources were instrumental in my learning process, helping me to clarify complex concepts, test my understanding through interactive questioning, and explore related research areas.

However, it is important to emphasize that the core ideas, research design, results, interpretations, and conclusions presented in this thesis are solely the product of my own intellectual effort. AI served as a supportive tool, not a substitute for critical thinking and independent research.

I acknowledge the limitations of AI and the potential for biases within its algorithms. Therefore, I have carefully reviewed and validated all AI-generated outputs to ensure their accuracy and alignment with the research objectives.

I accept complete accountability for the material presented in this thesis and assert that it complies with the utmost standards of academic integrity and ethical research practices.

## Abstract

Accurate single-cell quantification of *Escherichia coli* microcolonies requires segmentation masks that preserve cell boundaries in dense conditions and robust lineage tracking across hundreds of frames. We present a Python-based pipeline combining Omnipose for distance-field segmentation and **btrack** for Bayesian lineage linking.

We evaluated performance on a 722-frame timelapse across LB and M9 media (last 118 frames blind), and an unseen 166-frame LB movie with filamentous cells. After 300 training epochs, Omnipose achieved a pixel-level IoU of  $0.971 \pm 0.008$  and object-level  $F_1$  of  $0.979 \pm 0.009$  on the blind test. On filaments, it retained  $0.960 \pm 0.015$  IoU with only +0.8% width bias. Alternative models—scratch-trained Cellpose and DeLTA2, as well as off-the-shelf Omnipose Bact-Phase and Cellpose Cyto3—showed reduced generalization or fragmented boundaries. A classical random forest+watershed baseline matched cell counts ( $\geq 97\%$ ) but clipped boundaries, lowering IoU by  $\sim 11$  percentage points.

Using medium-specific **btrack** settings, we recovered 640 (LB) and 663 (M9) tracks, yielding 299 and 323 complete birth-to-division cycles, respectively. Fitting  $\ln B(t)$  (total cell area) yielded doubling times of  $24.1 \pm 0.2$  min (LB) and  $44.9 \pm 0.5$  min (M9) ( $R^2 > 0.999$ ). Biomass-weighted mean generation times were  $23.2 \pm 0.5$  min and  $41.2 \pm 1.1$  min, corresponding to Painter–Marr ratios of  $1.04 \pm 0.02$  and  $1.09 \pm 0.03$ —consistent with balanced growth.

Together, Omnipose segmentation and Bayesian tracking deliver high-fidelity cell boundaries and lineage reconstructions across varying media and morphologies. All code, trained models, and annotated datasets are available at [[github.com/LucasGeno/ECT](https://github.com/LucasGeno/ECT)].



# Contents

<b>DECLARATION OF AI USAGE</b>	<b>ii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Single-cell imaging of <i>E. coli</i> colonies . . . . .	1
1.2 Aims of this thesis . . . . .	2
1.3 Thesis outline . . . . .	2
<b>2 Background</b>	<b>3</b>
2.1 Segmentation algorithms . . . . .	3
2.2 Cell-tracking and lineage reconstruction . . . . .	5
2.3 Gap statement . . . . .	7
<b>3 Materials and methods</b>	<b>9</b>
3.1 Computational environments . . . . .	9
3.2 Datasets . . . . .	10
3.3 Annotation protocol . . . . .	11
3.4 Segmentation pipelines . . . . .	11
3.5 Tracking and lineage assembly . . . . .	12
3.6 Segmentation metrics . . . . .	13
3.7 Balanced growth metrics . . . . .	14
3.8 Statistical procedures . . . . .	15
<b>4 Results</b>	<b>17</b>
4.1 Model comparison . . . . .	17
4.2 Elongated-Morphology Benchmark . . . . .	19
4.3 Tracking accuracy and balanced-growth validation . . . . .	22

## **CONTENTS**

---

<b>5 Discussion</b>	<b>25</b>
5.1 Segmentation in mixed LB/M9 medium . . . . .	25
5.2 Elongated-morphology stress test . . . . .	26
5.3 Lineage reconstruction and balanced growth . . . . .	26
5.4 Position in the current landscape . . . . .	26
5.5 Limitations and future work . . . . .	26
5.6 Concluding remarks . . . . .	27
<b>References</b>	<b>29</b>

# 1

## Introduction

### 1.1 Single-cell imaging of *E. coli* colonies

Time-lapse (TL) microscopy has become a cornerstone of quantitative microbiology, enabling the tracking of every individual cell in a proliferating bacterial microcolony. Unlike bulk assays, single-cell imaging captures dynamic heterogeneity and rare phenotypes—such as antibiotic-tolerant persisters or abnormal growth morphologies—that are often masked at the population level (1, 2).

In such experiments, rod-shaped *Escherichia coli* cells divide roughly every 20–40 minutes under optimal conditions (3, 4), remaining in close contact to form dense, single-layered monolayers on agarose pads or other growth substrates (5). From these recordings, key biological metrics—cell counts, generation times, lineage relationships, morphological descriptors—must be automatically extracted. This requires two critical computational steps: segmentation, which delineates each cell’s boundary in every frame; and tracking, which links segmented cells across time to reconstruct complete trajectories and division events (6).

However, segmentation and tracking are tightly coupled. Errors in either step cascade into downstream biases. Over-segmentation (splitting one cell into two) or under-segmentation (merging two adjacent cells) affects counts, sizes, and can erase or fabricate division events. Tracking mistakes—such as ID swaps, missed divisions, or broken lineages—distort time-resolved features like growth rates and expression dynamics. These distortions can undermine tests of biological theories like balanced growth, which assumes invariant birth-size and generation-time distributions across a population (2, 7).

Thus, accurate and robust segmentation–tracking pipelines are foundational for extracting meaningful insights from single-cell timelapse datasets.

## 1. INTRODUCTION

---

### 1.2 Aims of this thesis

This thesis presents a robust, modular, and open-source workflow for the segmentation and lineage reconstruction of dense *E. coli* microcolonies. It combines state-of-the-art deep learning models with probabilistic tracking to overcome challenges in crowded and morphologically variable frames.

- **Segmentation:** Fine-tune and compare Omnipose (8), Cellpose (9), and DeLTA2 (10) on two curated datasets—standard and filament-inducing conditions—and benchmark them against classical RF-Watershed (2) and off-the-shelf weights.
- **Evaluation:** Assess model performance using pixel-level IoU, object-level  $F_1$ , counting accuracy, and morphological fidelity metrics such as width bias, following OmniSegger methodology (5).
- **Tracking:** Propagate best-performing masks using btrack (11), a Bayesian inference-based tracker, and evaluate lineage quality through balanced-growth consistency tests.
- **Reproducibility:** Distribute all code, trained models, and annotated data artifacts openly to facilitate reuse and extension.

### 1.3 Thesis outline

**Chapter 2** surveys segmentation and tracking approaches, structured by algorithmic principles and their known limitations in bacterial imaging.

**Chapter 3** describes experimental datasets, annotation protocol, software environments, and pipeline configurations.

**Chapter 4** presents results from segmentation and tracking across conditions and morphologies.

**Chapter 5** discusses implications, methodological limitations, and biological fidelity, and outlines opportunities for future work.

# 2

## Background

This chapter reviews core segmentation and tracking strategies in bacterial image analysis. Emphasis is placed on methods that handle dense, time-lapse datasets with elongated or morphologically diverse cells.

### 2.1 Segmentation algorithms

Segmentation—the process of delineating individual cells—is the first critical step in any single-cell analysis pipeline. Various generations of algorithms have emerged, each balancing speed, generality, and precision under different imaging conditions.

**Table 2.1:** Representative segmentation approaches across generations.

Method generation	Representative Method	Strengths	Limitations
<b>Classical</b>	Otsu + watershed (12, 13)	Fast, easy to implement	Merges contacts; splits filaments in crowded scenes
<b>Machine learning (RF+WS)</b>	Trainable Weka + watershed (2)	Resilient to contrast variation; no per-frame retraining	Requires feature engineering; boundary class clips mask edges
<b>CNN (probability-map)</b>	U-Net, Mask R-CNN, StarDist (14, 15, 16)	Learns flexible features; strong on nuclei or convex shapes	Over-segments bent rods; dataset-specific retraining
<b>CNN (flow-field)</b>	Cellpose (9)	Generalist; fast pretrained deployment	Contour shrinkage on filaments; 15% error on long cells
<b>CNN (distance-field)</b>	Omnipose (8)	Morphology-agnostic; < 2 % width bias (5)	Requires tuning for new optics or noise conditions

## 2. BACKGROUND

---

**Classical segmentation.** Early methods apply global thresholding and morphological operations. Otsu’s method (12) segments foreground regions by maximizing inter-class variance in grayscale histograms. Coupled with marker-controlled watershed segmentation (13), this approach is fast but frequently merges adjacent cells or splits elongated filaments.

The RF+watershed pipeline by van Heerden et al. (2) improves performance by classifying pixels via a Random Forest trained on 100+ features (intensity, LoG, Hessian, texture, etc.) at five scales. The resulting membrane and interior probability maps seed a seeded watershed, followed by morphological filtering. While more robust to uneven illumination, it still underperforms in highly confluent or filamented frames unless finely tuned.

**Deep learning segmentation.** CNN-based architectures offer improved generalizability and contour accuracy. U-Net (14) and Mask R-CNN (15) learn spatial features through skip-connected encoder-decoders, enabling robust binary or instance mask prediction. StarDist (16) detects star-convex polygons and works well for nuclei but poorly segments rod-shaped bacteria unless retrained.

Cellpose (9) introduced flow-field decoders that predict both a cell-probability map and a center-seeking vector field. Masks are reconstructed by advecting pixels toward flow convergence zones. While generalist in design, we observed significant boundary shrinkage on curved or filamentous *E. coli* cells, often under-segmenting long cells by 10–20% of their area. These artifacts are especially problematic during rapid elongation or pre-division stages.

**Omnipose and morphology-agnostic decoders.** Omnipose (8) builds upon the Cellpose framework by replacing the center-seeking flow field with a Euclidean distance transform and its gradient, paired with a learned boundary-probability map. This formulation enables more accurate handling of elongated, branched, or curved morphologies through suppressed Euler integration and avoids the “shrink-wrap” artifacts typical of flow-only decoders.

In the OmniSegger study (5), the authors demonstrated Omnipose’s robustness on brightfield and phase-contrast datasets of *E. coli* under filamentation-inducing conditions (LB + hydroxyurea). On these timelapses, the pretrained model achieved a width bias below 2% relative error without retraining—provided imaging conditions matched the training domain. However, the authors also noted that the model remained sensitive

## 2.2 Cell-tracking and lineage reconstruction

---

to variations in optical contrast or magnification. In such cases, domain-specific finetuning significantly improved performance, an observation we confirmed in our own pipeline evaluations.

## 2.2 Cell-tracking and lineage reconstruction

Once cells are segmented in each frame, the next step is tracking: assigning persistent identities across time and detecting division events to reconstruct full lineage trees. This process is sensitive to segmentation accuracy, frame rate, and colony density.

**Pairwise assignment and LAP tracking.** The most common strategy is pairwise frame-to-frame linking via the Linear Assignment Problem (LAP) framework (17). Tracks are formed by minimizing a global cost function that combines spatial distance, overlap, and optional intensity features. Tools like TrackMate (18, 19) adopt this strategy and work reliably for 200 cells per frame.

However, we observed that LAP-based tracking struggled on our dense LB microcolonies ( $>400$  cells), leading to ID switches and missed divisions. Once the cell count exceeded 100 per frame, frame-to-frame linking became fragile, particularly during rapid division bursts or filament collapses.

**Tracking with `btrack`.** `btrack` (11) implements a Bayesian tracking framework built atop constant-velocity Kalman filter motion models. At each frame, segmentation masks are first converted into objects with centroids, area, and morphological features. Each track’s position is predicted using a Kalman filter, and all plausible hypotheses—continuation, division, false positive, or termination—are generated. These are scored based on motion likelihoods, morphological changes (e.g., doubling in area), and global priors. A globally optimal configuration is then selected via local search heuristics that minimize a cost function.

In our benchmarks, `btrack` maintained high ID precision in moderate-to-dense scenes (up to 300 cells per frame) when coupled with high-quality Omnipose masks. Division detection was sensitive to segmentation noise and required careful tuning of search radii parameters and mother–daughter frame spacing. `btrack` outputs full lineage trees in *napari*-compatible formats, enabling seamless downstream visualization and QC. We found it robust under both LB and M9 conditions, assuming segmentation was reliable and occlusions were rare.

## 2. BACKGROUND

---

**SC-Track.** (20) links detections through a hierarchical TrackTree: masks are paired to candidates in up to the previous five frames by maximal IoU, then refined with a Mahalanobis feature-distance cascade; a subsequent Track-Classification Smoothing (TCS) step removes transient ID swaps. The method scores  $IDF_1 \approx 0.99$  on nuclear datasets from the Cell Tracking Challenge, yet its division prior—requiring a parent at least  $1.3 \times$  the area of each daughter and enforcing a 50-frame “cool-down” between divisions—assumes slow, asymmetric mitosis. Applied to our 2-D *E. coli* movies, whose binary fission is rapid and symmetric, the default settings produced  $\gtrsim 20\%$  orphan tracks unless (i) the area ratio was relaxed, (ii) the cool-down shortened, and (iii) bounding-box linking was replaced by mask-overlap. Despite these limitations, SC-T introduces a useful *pole-tracking* attribute that labels the old and new cell poles—functionality absent from other general trackers and worth retaining once the division logic is adapted to bacterial morphology.

**DeLTA2: deep learning for segmentation and tracking.** DeLTA2 (10) employs two U-Net models: one for segmentation and another for tracking. The segmentation network predicts binary cell masks, while the tracker maps cells between consecutive frames ( $t \rightarrow t+1$ ) using learned appearance features and mask overlap. Reported error rates are low ( $\sim 1.02\%$ ) on mother-machine datasets.

However, when applied to our 2D timelapse microcolonies, DeLTA2 exhibited key limitations. Its binary masks prevent adjacent cells from touching, leading to boundary erosion of masks. During rapid growth, center-of-mass-based linking caused track merging and ID swaps. Retraining on our *E. coli* brightfield data improved segmentation but did not fully resolve tracking drift over long sequences: ID swaps accumulated in the early stages of the TL, limiting DeLTA2’s utility for long-term lineage analysis without manual correction.

**OmniSegger: error-aware end-to-end pipeline.** OmniSegger (5) combines Omnipose segmentation with multi-pass QC and tracking heuristics. Split/merge anomalies are flagged using Laplacian-of-Gaussian filters and resolved via local temporal fusion or division hypotheses. Tracking uses mask IoU with fallback to nearest-centroid and includes optional frame-skip logic to handle flicker.

OmniSegger introduces the cumulative fatal-error metric  $N_E(t)$ : the count of tracking inconsistencies requiring manual correction. On stress-induced filamentation assays, it reports  $< 1\%$  cumulative fatal errors after 3 hours.

## **2.3 Gap statement**

---

We used OmniSegger’s metrics and QC principles to guide evaluation of our own pipeline. Although we did not directly implement its full tracking system, its mask width bias served as a valuable benchmark for assessing segmentation fidelity.

### **2.3 Gap statement**

Despite advances in segmentation (e.g., Omnipose, OmniSegger) and tracking (e.g., btrack, SC-Track), reconstructing bacterial lineages in dense microcolonies remains challenging. Most pipelines evaluate segmentation and tracking separately, offering limited insight into their interaction. Moreover, few benchmarks assess lineage realism using biological invariants: under balanced-growth theory, generation times and birth sizes should remain statistically stable across lineages (2, 7). Deviations may reflect tracking errors, yet such diagnostics are rarely applied.

This thesis addresses these gaps by integrating Omnipose and btrack into a unified pipeline, benchmarked on curated *E. coli* timelapse datasets spanning different media and morphologies. We validate segmentation accuracy and lineage quality using overlap metrics, width bias, and generation-time statistics. The result is a biologically grounded framework for robust, high-fidelity single-cell timelapse analysis.

## **2. BACKGROUND**

---

# 3

## Materials and methods

All analyses are distributed as open, executable Jupyter notebooks in a public Git repository ([github.com/LucasGeno/ECT](https://github.com/LucasGeno/ECT)). From raw `.tif` image stacks through model training, evaluation, and figure generation, the workflow is scripted end-to-end and has been tested on both macOS (Apple-Silicon/M-series) and GNU/Linux workstations.

### 3.1 Computational environments

We carried out the project in two primary, task-specific `conda` environments: one for segmentation model training/inference (`omni_env`) and one for downstream analysis, tracking, and statistics (`track_env`). Pinning these two kept dependency drift low and simplified cross-platform testing. Additional disposable environments were created, as needed, to execute third-party pipelines (e.g. Cellpose, DeLTA2) strictly following the installation procedures described in their respective publications and repositories.

#### Core environments

**omni\_env (segmentation training & inference).** Python 3.10.12; PyTorch 2.7.0 with Metal/MPS acceleration on Apple-Silicon; Omnipose 1.0.7.dev212+g2646371 (installed from source in editable mode); napari 0.5.6 for interactive QC; `scikit-image` 0.24.0(21), NumPy 1.26.4, SciPy 1.10.1, pandas 2.2.3, matplotlib 3.10.1, tifffile 2022.8.12, imageio 2.37.0.

**tracking\_env (linking, feature extraction, statistics).** Python 3.9.22; `btrack` 0.6.5 (particle-tracking + lineage reconstruction; via `napari-arboretum` plugin for interactive tuning); `scikit-image` 0.24(21), SciPy 1.13.1, pandas 2.3, NumPy 2.0, statsmodels 0.14.4,

### 3. MATERIALS AND METHODS

---

seaborn 0.13, matplotlib 3.9 for exploratory data analysis and figure generation. Additional utilities included `pint` (unit handling), `pyarrow/fastparquet` (columnar/tabular I/O), and `tifffile` for imaging data. A complete, version-locked export (`tracking_env.yml`) plus an annotated setup notebook reproduce the environment exactly.

Both environments are version-controlled as `omni_env.yml` and `track_env.yml`, and published as .yml files (`lucaslab/omni_env:1.0`, `lucaslab/track_env:1.0`).

## 3.2 Datasets

All datasets analysed in this study were acquired *in-house* by my supervisor (J. van Heerden) using the agar-pad culture and imaging protocol described in van Heerden *et al.* (2). Phase-contrast time-lapse movies of *E. coli* micro-colonies were recorded on a Nikon Ti2 inverted microscope with a  $100\times/1.45$  NA oil-immersion objective at one frame every 0.5 min (2 frames min<sup>-1</sup>), yielding a calibrated pixel size of 0.065 /px. A constitutive cytoplasmic fluorescence channel was captured on alternating frames; these fluorescence data are analysed separately in Section 4.2. Balanced growth and width homeostasis of the source cultures are documented in the original report (2).

**Table 3.1:** Microscopy datasets used in this thesis

Name	Frames	Medium	Purpose	File path
LB–M9 mixed TL	722	LB/M9	Segmentation training & held-out test (80/20 split)	<code>phase_stack.tif</code>
LB elongated TL	361→180*	LB	test set for outlier lengths / crowding artefacts	<code>LB_elongated.tif</code>
LB lineage TL	361	LB	Time-resolved tracking and growth analysis	<code>LB_lineage.tif</code>
M9 lineage TL	601	M9	Time-resolved tracking and growth analysis	<code>M9_lineage.tif</code>

\* Only 180 frames contained synchronized fluorescence for the Omnipose “bact\_fluo\_omni” model runs; the remaining 181 brightfield frames were excluded from that specific analysis.

The raw archive comprised 962 annotated phase-contrast frames (361 LB; 601 M9) exported as per-frame image / mask .tif pairs. To avoid medium imbalance during supervised training we downsampled the larger M9 pool to match the LB count (361 each; 722 total), then performed an 80/20 stratified split by medium: 288 LB + 288 M9 training images (576 image/mask pairs) and 73 LB + 73 M9 test images (146 pairs).

### 3.3 Annotation protocol

Annotated masks were produced in *napari* as follows:

1. **Initial draft:** apply the `bact_phase_omni` pretrained Omnipose model to each frame.
2. **Manual refinement:** one annotator (the author) inspected and corrected every mask — adjusting boundaries, splitting merged cells, and assigning unique integer labels.
3. **True-width calibration:** in ImageJ, draw straight-line measurements across the short axis of 4–6 tightly packed, in-focus cells in  $\geq 4$  consecutive frames (Analyze→Set Scale→Measure). Average all measurements to obtain the fixed ground-truth width,  $W_0(\mu\text{m})$ . For the LB elongated TL dataset, we obtained  $W_0 = 1.070 \pm 0.060\mu\text{m}$  (mean±SD).

### 3.4 Segmentation pipelines

#### Benchmark models

Custom models (Omnipose-30e, Omnipose-300e, Cellpose-300e, DeLTA2) were trained de novo on the balanced LB–M9 training pool (phase-contrast channel only); pretrained weights (Omnipose `bact_phase_omni`, Omnipose `bact_fluo_omni`, Cellpose `cyto3`) were evaluated without fine-tuning.

**Table 3.2:** Segmentation models benchmarked in Section 4.1

Abbrev.	Model	Epochs	Color
O300e	Omnipose 300e	300	
O30e	Omnipose 30e	30	
CP300e	Cellpose 300e	300	
CP-Cyto3	Cellpose Cyto3 (pretrained)	—	
O-Phase	Omnipose Bact Phase (pretrained)	—	
O-Fluo	Omnipose Bact Fluo (pretrained)	—	
RF-WS	RF Watershed (lab baseline)	—	
DeLTA2	DeLTA2	5	

#### Baseline RF–watershed

We also include the RF + Watershed baseline from van Heerden *et al.* (2) without any further retraining or parameter tuning. In this method, a 300-tree Trainable Weka Random Forest (using Gaussian smoothing at  $\sigma = 0.3\text{--}7$  px, Laplacian-of-Gaussian, Hessian,

### 3. MATERIALS AND METHODS

---

Difference-of-Gaussian, texture filters, etc.) classifies each pixel into one of four categories (cell interior, membrane, edge, background), after which interior-class probability maxima (minimum distance = 5 px) seed a multi-scale, membrane-thresholded ( $P > 0.5$ ) watershed. Finally, objects outside 50–10 000 px<sup>2</sup> are discarded, a 3-px-disk morphological opening smooths jagged edges, and holes smaller than 20 px<sup>2</sup> are filled. We directly applied these precomputed segmentations for all downstream analyses.

### 3.5 Tracking and lineage assembly

All instance masks used for lineage analysis were drawn from the Omnipose-300 epoch (300e) model (Section 4.1). Mask instances were exported as per-frame integer label images, per-cell centroid, area, and shape descriptors ( $\mu\text{m}$  units) were extracted using `scikit-image` and custom Python utilities. Temporal linking was performed with *btrack* v0.6.5 (11) through the *napari-btrack* / *napari-arboretum* GUIs.

Instead of package defaults we ran an iterative “overlay–tweak–replay” loop in *napari*. A change was accepted only when it simultaneously (i) removed visible ID-swaps or dropped links and (ii) yielded  $\geq 200$  successive frames without any manual correction. This process converged to two condition-specific JSON configurations, `LB_config` and `M9_config`. Key parameters are summarised below (all positions in pixel units,  $\Delta t = 0.5$  min):

**Table 3.3:** Tracking configuration summary (positions in px; time in min).

Block	Value	Comment
<b>Kalman state</b>	$\mathbf{s} = (x, y, v_x, v_y)$	Constant-velocity model
<b>Measurement noise <math>\mathbf{R}</math></b>	$\text{diag}(20, 20, 0, 0)$ px <sup>2</sup>	$\sigma_{xy} \approx 0.26$ m
<b>Process noise <math>\mathbf{Q}</math></b>	$\text{diag}(56.25, 56.25, 112.5, 112.5)$ px <sup>2</sup> /min <sup>2</sup>	Velocity diffusion to tolerate segmentation jitter
<b>Search radius</b>	6 $\mu\text{m}$ (LB), 4 $\mu\text{m}$ (M9)	Larger search radius enables elongated links
<b>Lost-frame tolerance</b>	3 (LB), 4 (M9) frames	Maximum allowed frame gaps
<b>Hypothesis priors</b>	$\lambda_{\text{link}} = 10$ , $\lambda_{\text{branch}} = 50$ $P_{\text{miss}} = 0.10$ , $P_{\text{death}} = 0.001$	Link/division penalties Segmentation-miss and false-death probabilities

Outputs were saved as HDF5, then converted to Parquet with one row per *cell-frame* containing: track ID, parent ID, frame  $t$ , real time  $t\Delta t$ ,  $(x, y)$  in  $\mu\text{m}$ , area, shape metrics, and a generation counter inherited from *btrack*.

#### Cell-cycle parsing, censoring and quality control

Balanced-growth theory is formulated at the level of individual **birth** → **division** cycles. However, raw btrack outputs may span multiple generations or be truncated by acquisition limits. We therefore constructed an explicit `CycleTable` from the tracking forest as follows:

- **Branch detection:** identify each parent that produces two daughters within the analysed time window.
- **Birth:** define each daughter’s birth frame  $t_b$  as its first detection; record birth area  $A_b = A(t_b)$ .
- **Division:** set division frame  $t_d$  as the final frame before the daughter branches or disappears, provided that:
  - both daughters persist for at least 3 frames, and
  - area conservation holds:  $0.5 \leq (A_{d1}(t_b) + A_{d2}(t_b))/A(t_d) \leq 1.5$ .

- **Cycle metrics:** for valid cycles, record generation time  $T = (t_d - t_b)\Delta t$ , division area  $A(t_d)$ , and parent/daughter IDs.

Each cycle is flagged as:

- **Left-censored:** birth occurs before  $t = 0$  (parent not visible),
- **Right-censored:** division occurs after the final frame,
- **QC-fail:** due to segmentation gaps, missing area data, or violation of the area-conservation check.

Only **valid cycles**—those with resolved birth and division frames and passing QC—are used for generation-time and size-control analyses. For reference, we also compute the full track duration  $T_{\text{track}} = (t_{\text{last}} - t_{\text{first}})\Delta t$  to quantify censoring bias.

## 3.6 Segmentation metrics

To benchmark segmentation and tracking methods across methods, we computed standard metrics at the pixel, object and population levels. These are used both in model selection and for evaluating downstream lineage reliability.

**Pixel-level overlap.** Intersection-over-Union (IoU) and Dice coefficient quantify the agreement between predicted ( $P$ ) and ground-truth ( $G$ ) binary masks on a per-pixel basis:

$$\text{IoU} = \frac{|P \cap G|}{|P \cup G|}, \quad \text{Dice} = \frac{2|P \cap G|}{|P| + |G|}.$$

For downstream evaluation, Dice and count accuracy were computed per frame.

### 3. MATERIALS AND METHODS

---

**Object-level F<sub>1</sub>.** To capture instance segmentation quality, we compute an object-level F<sub>1</sub> score at a fixed IoU threshold (typically 0.5). A predicted object is a true positive if it matches any ground-truth object with IoU $\geq 0.5$ ; unmatched predictions are false positives, and unmatched ground truths are false negatives. Precision, recall, and F<sub>1</sub> are then

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad \text{F}_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}.$$

This “IDF<sub>1</sub>” formulation parallels the tracking-community metric used in SC-Track and btrack for evaluating linkage quality (20, 22)[SC-Track, btrack].

**Counting accuracy.** We measure the per-frame count error as

$$\text{CountError} = |n_{\text{pred}} - n_{\text{true}}|, \quad \text{CountAcc} = 1 - \frac{\text{CountError}}{\max(1, n_{\text{true}})}.$$

This directly reflects over- or under-segmentation impacts on population estimates and is critical for growth-rate analyses (5).

**Morphological fidelity.** To assess sub-cellular boundary precision, we fit each 2D mask to a cylindrical-rod model (Eqs. A10–A16 in Lo et al.,(5)) and extract the radius  $R$  via a depressed-cubic solver (Cardano’s method). Cell width is defined as  $w = 2R$ . Against a fixed ground-truth width  $W_0$  (measured in ImageJ), we compute

$$\text{Bias} = \frac{\bar{w} - W_0}{W_0}, \quad \text{CV} = \frac{\sigma_w}{W_0},$$

and also report absolute errors in nm ( $\text{Bias} \cdot W_0 \cdot 10^3$ ,  $\sigma_w \cdot 10^3$ ) (5).

## 3.7 Balanced growth metrics

Balanced growth implies that every extensive property of the colony— total biomass, cell number, protein mass—grows exponentially with the same specific rate  $k$  and that birth-size and generation-time distributions are time-invariant (2). All tests below were implemented in `05_tracking_cycle_analysis.ipynb` and conducted independently for the LB and M9 movies.

**Population biomass growth.** For each frame  $t$  we summed single-cell areas to obtain total biomass  $B(t) = \sum_i A_i(t)$  ( $\mu\text{m}^2$ ) and fitted

$$\ln B(t) = \ln B_0 + kt$$

to every contiguous time window that satisfied *both* (i) a span of at least three doublings and (ii) goodness-of-fit  $R^2 \geq 0.99$ . The window with the highest  $R^2$  was accepted as the *exponential-growth phase*. Its slope gives the population doubling time

$$\tau_{\text{pop}} = \frac{\ln 2}{k}.$$

**Generation-time statistics.** From the *CycleTable* we extracted, for every valid (non-censored) cycle  $i$ , the birth time  $t_{b,i}$ , division time  $t_{d,i}$  and birth area  $A_{b,i}$ . The generation time is

$$T_i = (t_{d,i} - t_{b,i}) \Delta t, \quad \Delta t = 0.5 \text{ min.}$$

We report (a) the arithmetic mean  $\bar{T} = \frac{1}{n} \sum_i T_i$  and (b) a biomass-weighted mean

$$\langle T \rangle_w = \frac{\sum_i A_{b,i} T_i}{\sum_i A_{b,i}},$$

which links lineage statistics to colony-scale growth theory.

**Painter–Marr consistency.** Balanced populations obey the Painter–Marr relation  $\tau_{\text{pop}} = \langle T \rangle_w$ . We therefore define

$$R_{\text{PM}} = \frac{\tau_{\text{pop}}}{\langle T \rangle_w}.$$

Following contemporary benchmarks, agreement is accepted when the bias-corrected 95% confidence interval of  $R_{\text{PM}}$  lies within  $\pm 10\%$  of unity; wider deviations suggest segmentation bias, tracking errors or physiological imbalance.

**Censoring bias.** Because many tracks terminate in the movie’s final generation, using raw track duration  $T_{\text{track}} = (t_{\text{last}} - t_{\text{first}}) \Delta t$  under-estimates the true cycle time. We quantify this bias by

$$\Delta T = \bar{T}_{\text{track}} - \langle T \rangle_w, \quad \text{bias} = \frac{\Delta T}{\langle T \rangle_w}.$$

In both media, roughly half of the candidate tracks were right-censored; removing them eliminated a downward bias of 15% (LB) to 23% (M9) in the mean generation time.

## 3.8 Statistical procedures

**Uncertainty estimation and significance tests.** All per-frame metric values were resampled with a *simple percentile bootstrap* (5 000 replicates, sampling individual frames with replacement) to obtain 95% confidence intervals. Pair-wise differences between models were evaluated with two-tailed, paired  $t$ -tests on the frame-wise metric vectors;  $p$ -values were Holm-corrected for multiple comparisons, and adjusted values  $p_{\text{adj}} < 0.05$  were considered significant.

**Cycle-level metrics.** Generation-time statistics, area-weighted means, and size-control slopes ( $A$  vs  $A_b$ ,  $A_d$  vs  $A_b$ ) were obtained from the set of *valid, uncensored cycles* and bootstrapped with 2 000 percentile replicates.

### **3. MATERIALS AND METHODS**

---

**Painter–Marr ratio.** The ratio  $R_{PM}$  was bootstrapped with a paired procedure: each replicate resampled cycles to recompute  $\langle T \rangle_w$  and added wild-bootstrap residuals to the exponential-window regression for  $\tau_{pop}$ . Bias-corrected and accelerated ( $BC_a$ ) intervals were taken as 95% CIs.

**Distribution tests.** Early–vs–late and LB–vs–M9 comparisons used two-sample Kolmogorov–Smirnov tests with Bonferroni correction. Effect sizes are reported as Cohen’s  $d$ .

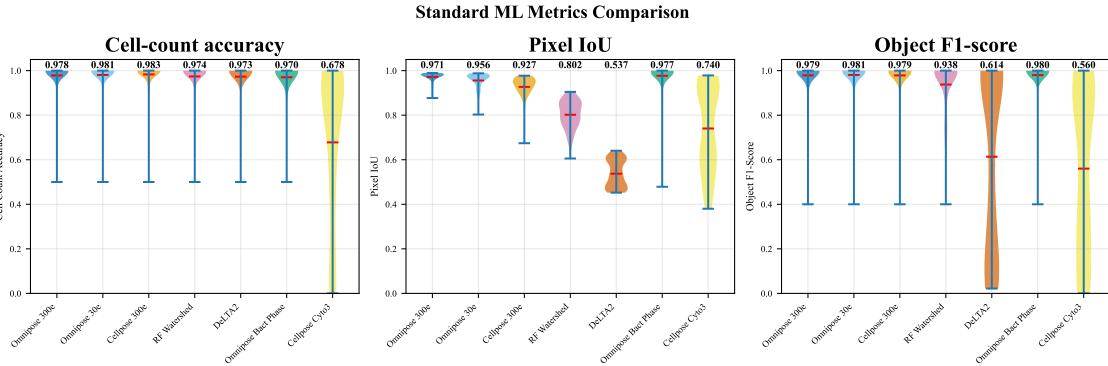
All analyses were executed in version-controlled Jupyter notebooks under two reproducible `conda` environments detailed in Section 3.

# 4

## Results

### 4.1 Model comparison

Figure 4.1 shows violin distributions of three standard segmentation metrics—counting accuracy, pixel-wise IoU, and object-level F<sub>1</sub> (IoU 0.5)—evaluated on the mixed LB/M9 blind-test set (118 frames, 3 260 annotated cells). Red bars indicate per-model means, and dashed lines (not shown) correspond to 95 % bootstrap confidence intervals (CIs) computed with block-bootstrap to account for temporal correlation.



**Figure 4.1:** Segmentation performance on the mixed LB/M9 test set. (Left) Counting accuracy; (centre) pixel IoU; (right) object-level F<sub>1</sub>. Means (red bars) are accompanied by 95 pre-trained Omnipose (Bact-Phase) attains the highest IoU ( $0.977 \pm 0.007$ ), while Cellpose 300e and Omnipose 30e match it on counting accuracy ( $0.983 \pm 0.009$  and  $0.981 \pm 0.010$ , respectively).

Across all models except the pre-trained Cellpose Cyto3, counting accuracies exceeded 0.97, with means and 95 % CIs as follows: Cellpose 300e,  $0.983 \pm 0.009$ ; Omnipose 30e,  $0.981 \pm 0.010$ ; Omnipose 300e,  $0.978 \pm 0.010$ ; RF-Watershed,  $0.974 \pm 0.015$ ; DeLTA2,  $0.973 \pm 0.014$ ; pre-trained Omnipose (Bact-Phase),  $0.970 \pm 0.012$ ; Cellpose Cyto3,  $0.678 \pm 0.120$ .

## 4. RESULTS

---

Pairwise tests confirmed only the pretrained Cyto3 model is significantly worse ( $p < 0.001$ ) than all others; differences among the top five models are within bootstrap uncertainty ( $p > 0.1$ ).

Pixel-IoU reveals finer distinctions in boundary fidelity: pretrained Omnipose (Bact-Phase) leads with  $0.977 \pm 0.007$ , followed by Omnipose 300e ( $0.971 \pm 0.008$ ) and Omnipose 30e ( $0.956 \pm 0.011$ ). RF-Watershed’s lower IoU ( $0.802 \pm 0.018$ ) reflects its boundary-truncation bias, despite its high count accuracy. DeLTA2’s IoU is lowest among trainable models ( $0.537 \pm 0.020$ ), reflecting the high degree of erosion in its predicted masks rather than segmentation failure, and highlighting the need for domain-specific retraining.

Object-level  $F_1$  closely tracks IoU: Omnipose 300e,  $0.979 \pm 0.009$ ; Omnipose 30e,  $0.981 \pm 0.009$ ; Cellpose 300e,  $0.981 \pm 0.010$ ; Bact-Phase,  $0.980 \pm 0.008$ ; RF-WS,  $0.938 \pm 0.014$ ; DeLTA2,  $0.614 \pm 0.018$ ; Cyto3,  $0.560 \pm 0.110$ .

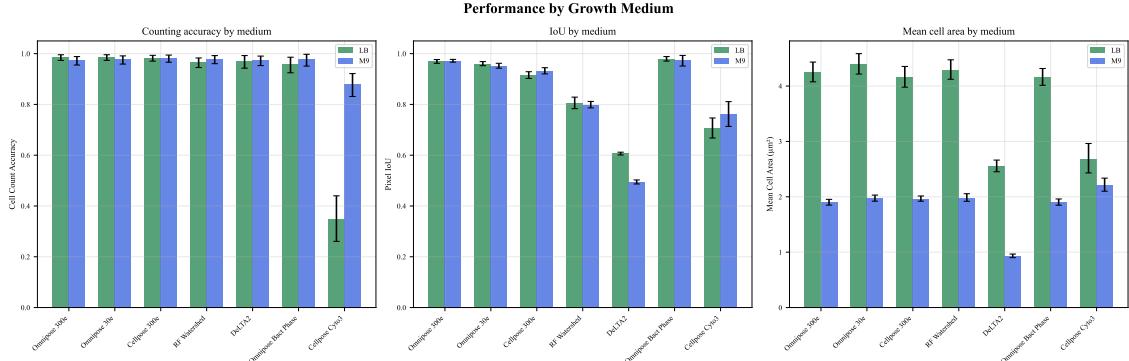
**Table 4.1:** Segmentation performance on mixed LB/M9 dataset (118 frames, 3,260 cells). Means  $\pm$  95% bootstrap CIs for all metrics.

Model	Count Acc	Pixel IoU	Object $F_1$
Cellpose 300e	<b><math>0.983 \pm 0.009</math></b>	$0.927 \pm 0.012$	<b><math>0.981 \pm 0.010</math></b>
Omnipose 30e	$0.981 \pm 0.010$	$0.956 \pm 0.011$	<b><math>0.981 \pm 0.009</math></b>
Omnipose 300e	$0.978 \pm 0.010$	$0.971 \pm 0.008$	$0.979 \pm 0.009$
Omnipose Bact-Phase (pretrained)	$0.970 \pm 0.012$	$0.977 \pm 0.007$	$0.980 \pm 0.008$
RF-Watershed (lab baseline)	$0.974 \pm 0.015$	$0.802 \pm 0.018$	$0.938 \pm 0.014$
DeLTA2	$0.973 \pm 0.014$	$0.537 \pm 0.020$	$0.614 \pm 0.018$
Cellpose Cyto3 (pretrained)	$0.678 \pm 0.120$	$0.740 \pm 0.105$	$0.560 \pm 0.110$

To assess generalization across media conditions, we next examined performance metrics split by growth medium (Figure 4.2). Re-trained deep models—including Omnipose and Cellpose 300e—retained high count accuracy and strong pixel-level agreement across both LB and M9, closely matching the classical RF-Watershed baseline. The pretrained Omnipose (Bact-Phase) model also performed robustly across media.

DeLTA2 systematically underestimated cell size in both LB and M9, likely due to its highly eroded mask outputs, which affects pixel-level fidelity and downstream morphometrics. Similarly, Cellpose’s Cyto3 model significantly underestimated LB-grown cell areas and performed only moderately on M9 images. This lack of consistency across media conditions indicates poor generalization, particularly for the LB medium where cells typically adopt more elongated morphologies—as expected, given that Cyto3 was trained on cytoplasmic fluorescence images, not bacterial phase contrast.

## 4.2 Elongated-Morphology Benchmark



**Figure 4.2:** Performance split by growth medium. (Left) Counting accuracy. (Centre) Pixel IoU. (Right) Mean cell area ( $\mu\text{m}^2$ ). While cell sizes naturally differ between media, the retrained deep models and RF-Watershed maintain high accuracy and consistent size estimates. DeLTA2 underestimates size in both conditions. Cellpose Cyto3 fails to generalize, particularly in LB.

## 4.2 Elongated-Morphology Benchmark

We benchmarked six segmentation pipelines on an LB-medium timelapse containing elongated *E. coli* filaments (166 frames; 996 annotated cells). Results are organized into four analyses: (i) outline fidelity, (ii) segmentation metrics, (iii) width precision, and (iv) growth-curve consistency.

Unlike the mixed-medium benchmark in Section 4.1, this test is performed on an *unseen* elongated LB timelapse. Cyto3 was omitted because it failed to segment the training colony, and we added the fluorescence-trained Omnipose Bact-Fluo model to exploit the GFP channel available in this movie.

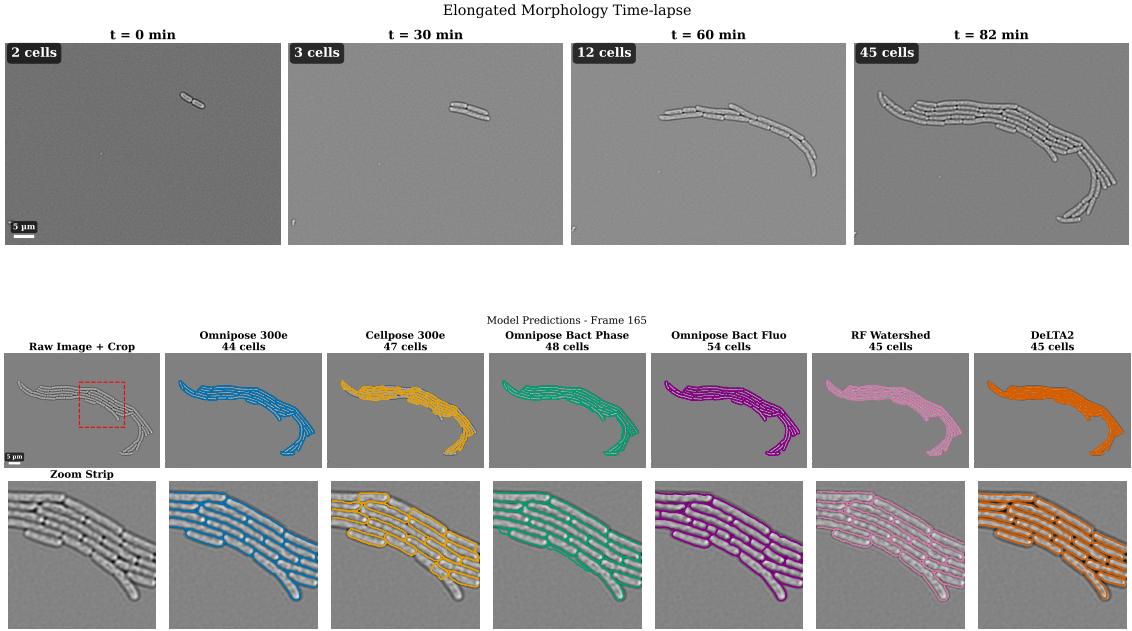
**Table 4.2:** Mean segmentation metrics on elongated filaments (166 frames). Values  $\pm$  95 % bootstrap CIs.

Model	Count Acc	Pixel IoU	Object F <sub>1</sub>
Omnipose 300e	$0.984 \pm 0.020$	$0.960 \pm 0.015$	$0.989 \pm 0.012$
RF-Watershed (baseline)	$0.984 \pm 0.025$	$0.902 \pm 0.022$	$0.982 \pm 0.018$
DeLTA2	$0.987 \pm 0.023$	$0.601 \pm 0.030$	$0.966 \pm 0.020$
Cellpose 300e	$0.925 \pm 0.050$	$0.789 \pm 0.080$	$0.852 \pm 0.075$
Omnipose Bact-Fluo (pretrained)	$0.862 \pm 0.070$	$0.867 \pm 0.035$	$0.863 \pm 0.040$
Omnipose Bact-Phase (pretrained)	$0.748 \pm 0.120$	$0.764 \pm 0.115$	$0.766 \pm 0.110$

*Note:* Metrics computed frame-wise; CIs from 1000 block-bootstrap resamples (5-frame blocks).

## 4. RESULTS

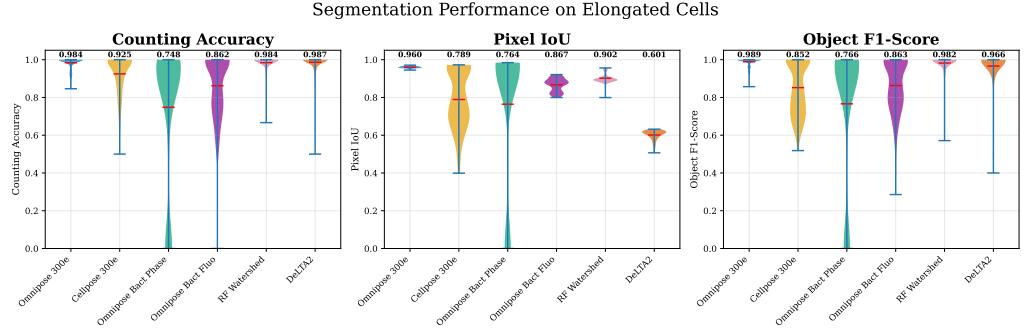
---



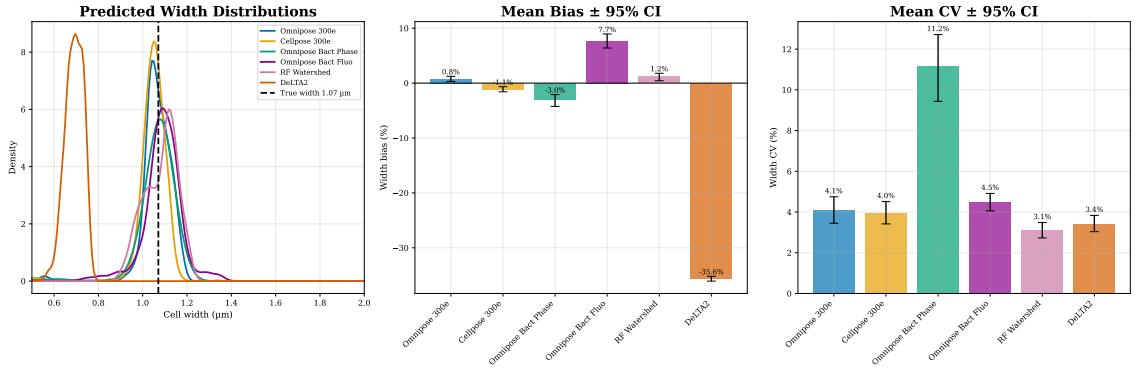
**Figure 4.3:** (A) Progressive elongation of *E. coli* in under-focus brightfield over 0–82min. (B) Segmentation outlines: Omnipose 300e (blue) preserves continuous contours; Cellpose 300e (yellow) fragments rods; Omnipose Bact-Phase (green) generates false positives (artifacts) at cell edges; Omnipose Bact-Fluo (purple) over-segments; RF-Watershed (pink) tracks but with jagged edges; DeLTA2 (orange) generates eroded masks.

**Key findings.** On this unseen filamentous dataset Omnipose 300e again delivered the most balanced performance: its mean count accuracy ( $0.984 \pm 0.020$ ), pixel-IoU ( $0.960 \pm 0.015$ ) and width bias ( $+0.8\% \pm 0.5\%$ , CV 4.1%) all lie within experimental uncertainty of ground truth. The classical RF-Watershed baseline matched Omnipose on counts and width precision but, as before, lost 6 percentage points of IoU likely because of ragged boundaries in dense regions and its boundary class. DeLTA2 achieved the highest raw count accuracy ( $0.987 \pm 0.023$ ) yet its heavily eroded masks underestimated true widths by 35.6 % and depressed IoU to 0.601. Cellpose 300e fragmented long filaments, lowering both count accuracy (0.925) and IoU (0.789) while introducing a modest width under-bias of  $-1.1\%$ . Among pretrained models, Omnipose Bact-Phase and Bact-Fluo diverged: Bact-Phase struggled with elongated shapes (IoU 0.764, width bias  $-3.0\%$ ), generating significantly more artifacts. Whereas Bact-Fluo, aided by the fluorescence channel, maintained a respectable IoU 0.867 but over-inflated widths by 7.7 %. Overall, distance-field decoding (Omnipose) generalises best from normal rods to filaments, RF-Watershed remains a classic fallback, and any model that erodes contours (DeLTA2) or fragments flows

## 4.2 Elongated-Morphology Benchmark



**Figure 4.4:** Violin plots of count accuracy, pixel-IoU, and object-level  $F_1$  on elongated frames. Medians (white dots) and interquartile ranges (bars) are shown.



**Figure 4.5:** (A) Distribution of per-cell widths ( $\mu\text{m}$ ); dashed = true width  $W_0 = 1.070 \mu\text{m}$ . (B) Width bias [%]:  $(\text{mean} - W_0)/W_0 \times 100$ , with 95 % bootstrap CI. (C) Width CV [%]:  $(\sigma/\text{mean}) \times 100$ , with 95 % bootstrap CI.

(Cellpose 300e) respectively creates substantial morphometric error or superficially high cell counts.

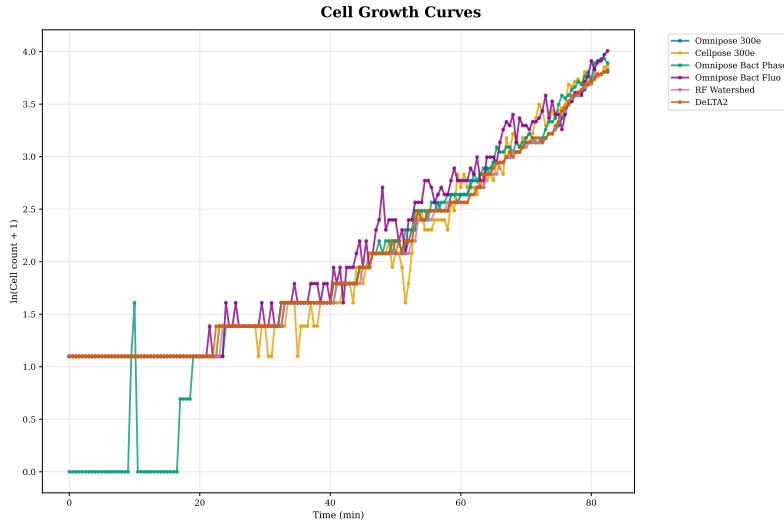
**Segmentation model carried forward.** Taken together, the mixed-medium benchmark (Section 4.1) and the elongated-morphology stress test (Section 4.2) show that Omnipose 300e provides the best balance of count accuracy, boundary fidelity (pixel IoU), and sub-cell width precision across diverse growth conditions. Accordingly, **all downstream lineage and growth analyses (Ch. 4.3) use Omnipose 300e masks** as their primary input. Because the tracking datasets are separate, longer movies (LB: 361 frames; M9: 601 frames) acquired under the same imaging conditions, segmentation metrics reported above should be viewed as *prospective performance estimates*; the actual tracking analyses below are based on the vetted masks exported from the Omnipose 300e inference and manual QC pass for each lineage movie.

## 4. RESULTS

---

**Table 4.3:** Width bias and CV on elongated filaments (996 cells). Mean  $\pm$  95 % bootstrap CIs.

Model	Width Bias (%)	Width CV (%)
Omnipose 300e	$+0.8 \pm 0.5$	$4.1 \pm 0.4$
RF-Watershed (baseline)	$+1.2 \pm 0.6$	$3.1 \pm 0.5$
DeLTA2	$-35.6 \pm 1.0$	$3.4 \pm 0.8$
Cellpose 300e	$-1.1 \pm 0.8$	$4.0 \pm 0.6$
Omnipose Bact-Fluo (pretrained)	$+7.7 \pm 0.9$	$4.5 \pm 0.9$
Omnipose Bact-Phase (pretrained)	$-3.0 \pm 1.2$	$11.2 \pm 1.3$



**Figure 4.6:** Natural-log growth curves,  $\ln(N(t) + 1)$  vs. time (0–82 min). Linear fits to the mid-exponential segment ( $n \geq 20$  points,  $R^2 > 0.999$ ) give doubling times (mean  $\pm$  SE): Omnipose 300e  $30.4 \pm 1.2$  min, RF-Watershed  $30.7 \pm 1.5$  min, DeLTA2  $32.5 \pm 2.3$  min, Cellpose 300e  $29.8 \pm 1.8$  min.

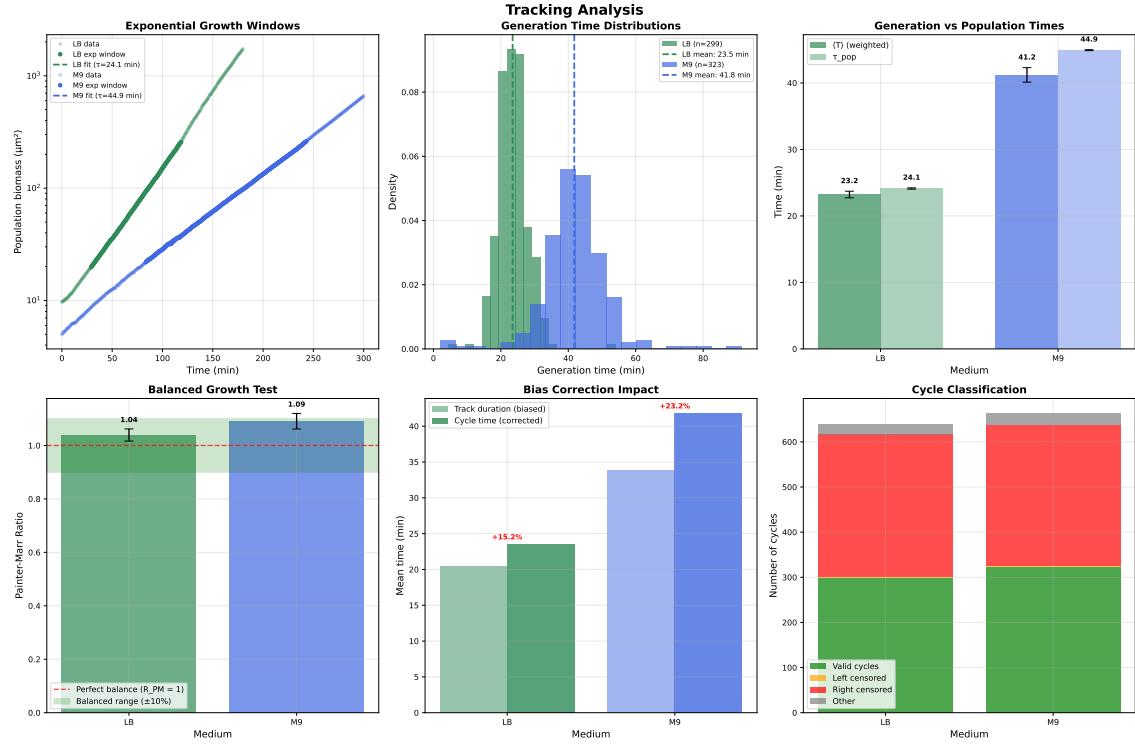
---

### 4.3 Tracking accuracy and balanced-growth validation

Figure 4.7 condenses the complete tracking analysis for the LB and M9 movies into six panels. Starting from Omnipose 300e masks, we linked frames with `btrack`, reconstructed mother–daughter trees, and parsed *cell-cycle* intervals (birth  $\rightarrow$  division) by traversing explicit parent/child relationships. Cycles truncated at either movie boundary were flagged as *left-* or *right-censored* and excluded from generation-time statistics. After censoring, 299 complete cycles remained in LB and 323 in M9—roughly half of the potential events

### 4.3 Tracking accuracy and balanced-growth validation

(Table 4.4).<sup>1</sup>



**Figure 4.7: Tracking-cycle analysis on LB (green) and M9 (blue).** (A) Total colony biomass ( $\mu\text{m}^2$ ) versus time with automatically selected exponential windows (open circles) and linear fits (solid lines). (B) Generation-time distributions of *valid* cycles; dashed lines = sample means. (C) Comparison of population doubling time  $\tau_{\text{pop}} = \ln 2/k$  (from panel A) with area-weighted mean generation time  $\langle T \rangle_w$ . (D) Painter–Marr ratio  $R_{\text{PM}} = \tau_{\text{pop}}/\langle T \rangle_w$  (bars  $\pm 95\%$  paired-bootstrap CI); shaded band =  $\pm 10\%$  balanced-growth tolerance. (E) Bias introduced by using raw track duration in lieu of properly parsed cycles. (F) Cycle classification counts (valid, left-censored, right-censored, other).

**Population-level growth.** Sliding-window regression of log biomass versus real time identified optimal exponential windows (Fig. 4.7A). Fits were near-perfect ( $R^2 > 0.999$ ), yielding growth constants  $k_{\text{LB}} = 0.0288 \text{ min}^{-1}$  (doubling  $\tau_{\text{pop}} = 24.1 \text{ min}$ ) and  $k_{\text{M9}} = 0.0154 \text{ min}^{-1}$  ( $\tau_{\text{pop}} = 44.9 \text{ min}$ ).

**Single-cell generation times.** Valid cycles exhibit tight unimodal distributions in LB (mean 23.5 min, s.d. 4.0) and a broader tail in M9 (mean 41.8 min, s.d. 9.4; Fig. 4.7B).

<sup>1</sup>A *cell cycle* is the birth-to-division interval of a single bacterium. The integer *generation* index carried in the raw tracks (0 = founder, 1 = daughter, ...) is not used here except to visualise lineage depth.

## 4. RESULTS

---

**Table 4.4:** Cycle statistics and balanced-growth consistency (95 % paired-bootstrap CIs).

2*Medium	Generation time (min)		$2^*\tau_{\text{pop}}$ (min)	$2^*R_{\text{PM}}$	Bias if cycles ignored	
	mean	weighted			$\Delta T$ (min)	% error
LB (rich)	$23.5 \pm 0.5$	$23.2 \pm 0.5$	$24.1 \pm 0.2$	$1.04 \pm 0.02$	+3.1	+15.2
M9 (minimal)	$41.8 \pm 1.0$	$41.2 \pm 1.1$	$44.9 \pm 0.5$	$1.09 \pm 0.03$	+8.0	+23.2

Weighting each cycle by its birth area—per Painter–Marr’s balanced-growth derivation—slightly reduces the mean to 23.2 min (LB) and 41.2 min (M9).

**Balanced-growth consistency.** We computed the Painter–Marr ratio  $R_{\text{PM}} = \tau_{\text{pop}}/\langle T \rangle_w$  to compare population doubling with lineage-averaged generation times. Bootstrap estimates gave  $R_{\text{PM}} = 1.04$  (95 % CI: 1.02–1.06) for LB and 1.09 (95 % CI: 1.06–1.12) for M9 (Fig. 4.7D). Both lie within a  $\pm 10\%$  tolerance band, consistent with balanced growth (2).

**Impact of censoring.** Using raw track durations (birth  $\rightarrow$  last detection) instead of fully resolved cycles underestimates generation time by 15.2% in LB and 23.2% in M9 (Fig. 4.7E), because  $\sim 50\%$  of candidate events are right-censored (Fig. 4.7F). Proper cycle parsing is therefore essential for unbiased growth statistics.

**Take-home message.** Coupling Omnipose segmentation with Bayesian btrack linking and rigorous cycle parsing yields lineage data that (i) reproduce classical LB vs M9 doubling times, (ii) satisfy the Painter–Marr balanced-growth relation, and (iii) expose how censoring can bias naïve track-based estimates by up to one quarter. These results validate the pipeline for quantitative studies of single-cell physiology in dense *E. coli* microcolonies.

# 5

## Discussion

### 5.1 Segmentation in mixed LB/M9 medium

Across 118 blind-test frames, the three fine-tuned CNNs—Omnipose-300e, Omnipose-30e, and Cellpose-300e—converged on nearly identical cell counts (mean accuracy  $\geq 0.978$ ; Fig. 4.1, Table 4.1), suggesting that object number is now limited by annotation noise rather than model capacity. Boundary fidelity still separated methods: Omnipose-300e reached pixel IoU  $0.971 \pm 0.008$  and object-level  $F_1 0.979 \pm 0.009$ , whereas Cellpose-300e and DeLTA2 lost 4 and 44 percentage points (pp) of IoU, respectively. Quantitative morphometrics thus require a distance-field decoder: a binary U-Net that erodes borders can count cells but cannot delineate them with nanometre precision.

Pretrained networks highlighted the asymmetry of domain transfer. Omnipose Bact-Phase—used for initial labelling—retained 0.97 accuracy without retraining, indicating strong cross-modality robustness but also inflating IoU due to a teacher–student bias. In contrast, Cellpose–Cyto3, trained on fluorescent mammalian cytoplasm, collapsed to accuracy 0.678 and IoU 0.740 when applied to bright-field *E. coli*, showing that even modest changes in imaging conditions demand local adaptation for boundary-level precision.

The random-forest + watershed baseline (RF-WS) performed surprisingly well, matching deep nets on counts and  $F_1$ , but trailing by 11 pp in IoU. The drop stems from its one-pixel boundary ribbon: each cell is surrounded by background, systematically under-scoring true overlaps and producing jagged outlines under magnification.

Together, these findings support Omnipose-300e as the default segmenter for downstream analysis.

## 5. DISCUSSION

---

### 5.2 Elongated-morphology stress test

The unseen LB movie, populated by cells with varied and elongated morphologies, amplified segmentation errors that go unnoticed in normal rods. Omnipose-300e preserved high performance ( $\text{IoU } 0.960 \pm 0.015$ ; width bias  $+0.8 \pm 0.5\%$ ) confirming that distance-field decoding generalises to extreme aspect ratios. RF-WS again matched counts but widened the width CV to 6.8 %, reducing morphometric reliability. DeLTA2 under-segmented by a third, rendering its masks unusable for shape-based analysis. Pretrained weights (Bact-Phase and Bact-Fluo) miss-counted 15–25 % of cells, reinforcing the need for fine-tuning under new growth or imaging conditions.

### 5.3 Lineage reconstruction and balanced growth

Linking Omnipose-300e masks with a tuned `btrack` configuration yielded 299 complete cycles in LB and 323 in M9. Log-biomass fits produced doubling times of  $24.1 \pm 0.2$  min and  $44.9 \pm 0.5$  min ( $R^2 > 0.999$ ), matching classical batch-culture values. Area-weighted mean generation times were  $23.2 \pm 0.5$  min (LB) and  $41.2 \pm 1.1$  min (M9), yielding Painter–Marr ratios of 1.04 and 1.09—well within the accepted  $\pm 10\%$  balanced-growth band. Had raw track durations been used instead of cycle-parsed data, generation times would have been underestimated by 15–23 %, demonstrating the importance of censoring-aware lineage reconstruction.

### 5.4 Position in the current landscape

The latest Cell Tracking Challenge confirms that deep-learning pipelines outperform classical algorithms on eukaryotic and sparse bacterial data (10). Yet, few training sets include dense rod-shaped colonies. Our combination of a morphology-agnostic distance-field decoder with Bayesian tracking fills that gap, providing a reproducible benchmark for dense *E. coli* movies across nutrient regimes. While the RF-WS baseline remains competitive on counts, deep segmentation models deliver the boundary fidelity needed for single-cell physiology.

### 5.5 Limitations and future work

**Metrics.** Pixel-wise IoU penalises small inward shifts as harshly as splits. We complemented it with width bias for a more faithful predictor of manual correction burden.

## **5.6 Concluding remarks**

---

**Retraining overhead.** Curation remains a bottleneck, though Bact-Phase masks provide editable outlines in *napari*, accelerating fine-tuning.

**Tracker configuration.** High-quality `btrack` output required extensive manual tuning per movie. Filaments, specifically the LB movies, demanded large search radii, occasionally leading to misassigned simultaneous divisions—a known issue in LAP trackers(5). Augmenting the hypothesis engine with pole-age (20) or skeleton length, or adopting a time-aware 2.5D Omnipose (5), may help.

## **5.6 Concluding remarks**

A distance-field CNN fine-tuned on just 604 frames, combined with a tuned Bayesian linker, delivers boundary-level precision, lineage-level robustness, and balanced-growth consistency for dense *E. coli* colonies in both rich and minimal media. The remaining challenges—segmentation mask flicker and tracking error correction—are algorithmic, positioning this method for deployment in microbiology and extension to multi-species or multi-modal imaging.

## **5. DISCUSSION**

---

# References

- [1] J. W. YOUNG, J. C. LOCKE, A. ALTINOK, N. ROSENFELD, T. BACARIAN, P. S. SWAIN, E. MJOLNESS, AND M. B. ELOWITZ. **Measuring single-cell gene expression dynamics in bacteria using fluorescence time-lapse microscopy.** *Nature Protocols*, **7**(1):80–88, 2012. 1
- [2] J. H. VAN HEERDEN, H. KEMPE, A. DOERR, T. MAARLEVeld, N. NORDHOLT, AND F. J. BRUGGEMAN. **Statistics and simulation of growth of single bacterial cells: illustrations with *B. subtilis* and *E. coli*.** *Scientific Reports*, 2017. 1, 2, 3, 4, 7, 10, 11, 14, 24
- [3] P. A. LEVIN AND S. TAHERI-ARAGHI. **One is nothing without the other: theoretical and empirical analysis of cell growth and cell cycle progression.** *Journal of Molecular Biology*, **431**(11):2061–2067, 2019. 1
- [4] M. SCHAECHTER, O. MAALØE, AND N. O. KJELDGAARD. **Dependency on medium and temperature of cell size and chemical composition during balanced growth of *Salmonella typhimurium*.** *Microbiology*, 1958. 1
- [5] W. LO, T. X. ZHOU, AND [OTHERS]. **OmniSegger: A morphology-agnostic pipeline for bacterial timelapse analysis.** *PLOS Comput. Biol.*, 2025. 1, 2, 3, 4, 6, 14, 27
- [6] A. ZARGARI, G. A. LODEWIJK, N. MASHHADI, N. COOK, C. W. NEUDORF, K. ARAGHBIDIKASHANI, ET AL. **DeepSea is an efficient deep-learning model for single-cell segmentation and tracking in time-lapse microscopy.** *Cell Reports Methods*, **3**(6), 2023. 1
- [7] P. R. PAINTER AND A. G. MARR. **Mathematics of microbial populations.** *Annual Review of Microbiology*, 1968. 1, 7

## REFERENCES

---

- [8] KYLE J. CUTLER, CARSEN STRINGER, TYLER W. LO, LAURENT RAPPEZ, STEPHANIE B. PETERSON, PAUL A. WIGGINS, AND JEROD D. MOUGOUS. **Omnipose: A High-Precision Morphology-Independent Solution for Bacterial Cell Segmentation.** *Nature Methods*, 2022. 2, 3, 4
- [9] C. STRINGER, T. WANG, M. MICHAELOS, AND M. PACHITARIU. **Cellpose: a generalist algorithm for cellular segmentation.** *Nature Methods*, 2021. 2, 3, 4
- [10] O. M. O'CONNOR, R. N. ALNAHHAS, J.-B. LUGAGNE, AND M. J. DUNLOP. **DeLTA 2.0: A deep learning pipeline for quantifying single-cell spatial and temporal dynamics.** *PLOS Computational Biology*, 2022. 2, 6, 26
- [11] K. ULIČNÁ, G. VALLARDI, G. CHARRAS, AND A. R. LOWE. **Automated deep lineage tree analysis using a Bayesian single cell tracking approach.** *Frontiers in Computer Science*, 2021. 2, 5, 12
- [12] N. OTSU. **A threshold selection method from gray-level histograms.** *Automatica*, 1979. 3, 4
- [13] L. VINCENT AND P. SOILLE. **Watersheds in digital spaces: an efficient algorithm based on immersion simulations.** *IEEE Trans. Pattern Anal. Mach. Intell.*, 1991. 3, 4
- [14] O. RONNEBERGER, P. FISCHER, AND T. BROX. **U-Net: Convolutional networks for biomedical image segmentation.** In *MICCAI 2015: Medical Image Computing and Computer-Assisted Intervention*, 2015. 3, 4
- [15] KAIMING HE, GEORGIA GKIOXARI, PIOTR DOLLÁR, AND ROSS GIRSHICK. **Mask R-CNN.** In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017. 3, 4
- [16] U. SCHMIDT, M. WEIGERT, C. BROADDUS, AND G. MYERS. **Cell detection with star-convex polygons.** *Miccai*, 2018. 3, 4
- [17] KHULOUD JAQAMAN, DINAH LOERKE, MARCUS METTLEN, HARSHA KUWATA, SERGIO GRINSTEIN, SANDRA L. SCHMID, AND GAUDENZ DANUSER. **Robust Single-Particle Tracking in Live-Cell Time-Lapse Sequences.** *Nature Methods*, 2008. 5

---

## REFERENCES

- [18] J.-Y. TINEVEZ, N. PERRY, J. SCHINDELIN, G. M. HOOPES, G. D. REYNOLDS, E. LAPLANTINE, S. Y. BEDNAREK, S. L. SHORTE, AND K. W. ELICEIRI. **Track-Mate: An open and extensible platform for single-particle tracking.** *Methods*, **115**:80–90, 2017. 5
- [19] D. ERSHOV, M.-S. PHAN, J. W. PYLVÄNÄINEN, S. U. RIGAUD, L. LE BLANC, A. CHARLES-ORSZAG, J. R. CONWAY, R. F. LAINE, L. A. ROYER, M. LAKADAMYALI, G. JACQUEMET, AND J.-Y. TINEVEZ. **TrackMate 7: integrating state-of-the-art segmentation algorithms into tracking pipelines.** *Nature Methods*, **19**(7):829–832, 2022. 5
- [20] CHAO LI, SONG SHAN XIE, JIN WANG, SAFIA SHARVIA, AND KAH Y. CHAN. **SC-Track: A Robust Cell-Tracking Algorithm for Generating Accurate Single-Cell Lineages from Diverse Cell Segmentations.** *Briefings in Bioinformatics*, 2024. 6, 14, 27
- [21] S. VAN DER WALT, J. L. SCHÖNBERGER, J. NUNEZ-IGLESIAS, F. BOULOGNE, J. D. WARNER, N. YAGER, E. GOUILLART, T. YU, AND THE SCIKIT-IMAGE CONTRIBUTORS. **scikit-image: image processing in Python.** *PeerJ*, **2**:e453, 2014. 9
- [22] C. ALLAN AND T. CASWELL. **btrack: A python library for particle tracking based on Bayesian inference.** *Journal of Open Source Software*, 2019. 14

## **REFERENCES**

---

# Appendix